



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Komputer

Kadir (2017:2) mengatakan bahwa komputer merupakan peralatan elektronik yang bermanfaat untuk melaksanakan berbagai pekerjaan yang dilakukan oleh manusia.

Krisbiantoro (2018:3) mengatakan bahwa komputer adalah sekumpulan alat elektronik yang saling bekerja sama, dapat menerima data (*input*), mengolah data (*process*) dan memberikan informasi (*output*) serta terkoordinasi dibawah control program yang tersimpan di memori komputer.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa komputer adalah alat elektronik yang berguna dalam mengolah hingga menjadi informasi untuk membantu pekerjaan manusia.

2.1.2 Perangkat Lunak

Perangkat Lunak, menurut Swara dan Pebriadi (2016:28), merupakan seluruh perintah yang digunakan untuk memproses informasi.

Menurut Puspita (2018:20), perangkat lunak merupakan program komputer, struktur data, dan dokumentasi yang berkaitan, yang menyediakan metode logika, prosedur atau control yang diminta.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa perangkat lunak adalah kumpulan dari data elektronik yang disimpan dan diatur oleh program komputer untuk memproses informasi.

2.1.3 Website

Abdulloh (2018:1) mengatakan bahwa *website* dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui



jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia.

Hutahaean dan Azhar (2018:21) mengemukakan bahwa *website* (situs web) merupakan kumpulan dari halaman-halaman web yang berhubungan dengan file-file lain yang terkait.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *website* adalah kumpulan halaman yang berisi file-file yang berhubungan dan dapat diakses oleh semua orang didunia menggunakan jaringan internet.

2.1.4 Data

Menurut Yana dan Hondro (2016:42), data adalah sesuatu yang belum mempunyai arti bagi penerimanya dan masih memerlukan adanya suatu pengolahan. Data bisa berwujud suatu keadaan, gambar, suara, huruf, angka, matematika, bahasa ataupun simbol-simbol lainnya yang bisa kita gunakan sebagai bahan untuk melihat lingkungan, obyek, kejadian ataupun suatu konsep.

Data menurut Jaluhu dan Trianovie (2020:2) adalah sekumpulan baris fakta yang mewakili peristiwa yang terjadi pada organisasi atau pada lingkungan fisik sebelum di olah kedalam format yang bisa dimengerti dan digunakan orang.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa data merupakan fakta atas suatu kejadian yang belum dapat dijadikan informasi berguna bagi pihak yang membutuhkan.

2.1.5 Basis Data

Abdulloh (2018:103) berpendapat bahwa *database* atau Basis data adalah kumpulan informasi yang disimpan dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi.

Rusmawan (2019:40) mengatakan bahwa basis data dapat didefinisikan sebagai himpunan kelompok data yang saling berhubungan dan diorganisasi sedemikian rupa sehingga kelak dapat dimanfaatkan dengan cepat dan mudah.

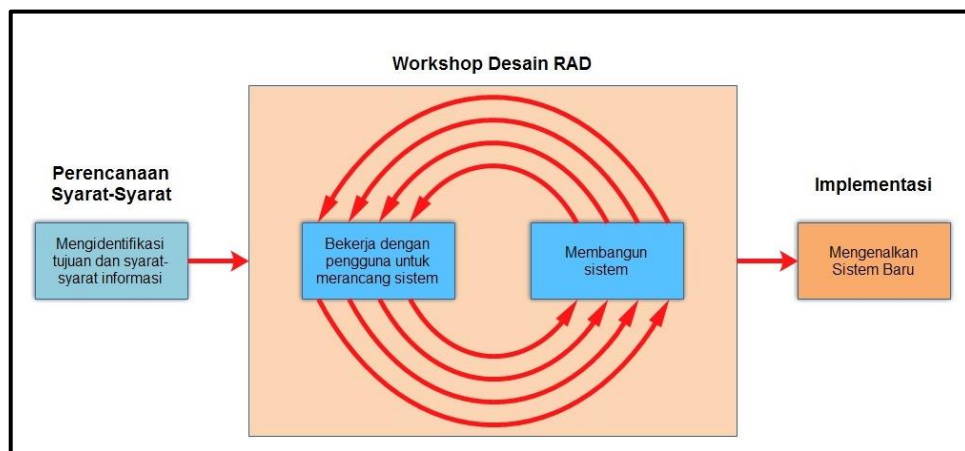


Berdasarkan pernyataan diatas, dapat disimpulkan bahwa basis data adalah kumpulan data yang saling terhubung dan disimpan dalam komputer sehingga dapat diperoleh kembali sebagai sebuah informasi.

2.1.6 Metode *Rapid Application Development* (RAD)

Menurut Sagala (2018:88), “*Rapid Application Development* (RAD) atau *rapid prototyping* adalah model proses pembangunan perangkat lunak yang tergolong dalam teknik inkremental (bertingkat)”.

Rapid Application Development (RAD) menekankan pada siklus pembangunan pendek, singkat, dan cepat. Waktu yang singkat adalah batasan yang penting untuk model ini. *Rapid Application Development* (RAD) menggunakan metode iterative (berulang) dalam mengembangkan sistem dimana *working model* (model kerja) sistem dikonstruksikan diawal tahap pengembangan dengan tujuan menetapkan kebutuhan (requirement) pengguna.



(Sumber : Kendall dan Kendall (2010) dalam Susilowati dan Negara (2018 : 26))

Gambar 2.1 Tahapan Metode *Rapid Application Development* (RAD)

Model RAD memiliki 3 tahapan sebagai berikut :

1. *Requirements Planning* (Perencanaan Syarat-Syarat)

Fase ini merupakan fase dimana pengguna dan penganalisis mengidentifikasi tujuan dari sistem yang akan dibangun serta mengidentifikasi syarat-syarat informasi yang akan timbul untuk mencapai



tujuan tersebut serta menganalisa semua sistem yang dibutuhkan oleh pengguna.

2. RAD Design Workshop (Workshop Desain RAD)

Fase ini adalah fase untuk merancang dan memperbaiki yang bisa digambarkan sebagai *workshop*. Penganalisis dan pemrogram dapat bekerja membangun dan menunjukkan representasi visual desain dan pola kerja kepada pengguna. *Workshop* desain ini dapat dilakukan selama beberapa hari tergantung dari ukuran aplikasi yang akan dikembangkan.

3. Implementation (Implementasi)

Fase ini penganalisis berkerja dengan para pengguna secara intensif selama workshop berlangsung, dan merancang beberapa aspek dan nonteknis yang dibutuhkan. Segera setelah aspek-aspek ini disetujui dan sistem-sistem dibangun dan disaring, sistem-sistem baru atau bagian dari sistem diujicoba dan kemudian diperkenalkan kepada organisasi.

2.2 Teori Judul

2.2.1 Aplikasi

Simargolang dan Nasution (2018:9) berpendapat bahwa Aplikasi adalah suatu program berbentuk perangkat lunak yang berjalan pada suatu sistem tertentu yang berguna untuk membantu berbagai kegiatan yang dilakukan oleh manusia.

Siregar, Siregar, dan Melani (2018:113) mengatakan bahwa secara umum aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya aplikasi merupakan suatu perangkat komputer yang siap pakai bagi user.

Dari pendapat diatas, dapat disimpulkan bahwa aplikasi adalah sebuah perangkat lunak yang berfungsi yang siap dipakai oleh pengguna untuk membantu menyelesaikan berbagai kegiatan yang dilakukan oleh manusia.

2.2.2 Persediaan Barang

Ramadhani, Suryadi, dan Irmayani (2018:36) mengemukakan bahwa persediaan merupakan barang-barang yang dimiliki untuk kemudian dijual atau



digunakan dalam proses produksi atau dipakai untuk keperluan non produksi dalam siklus kegiatan normal.

Baihaqi, Islamadina, dan Alfairuz (2018:11) mengatakan bahwa barang merupakan setiap benda baik yang berwujud maupun tidak berwujud, bergerak ataupun tidak bergerak, yang mempunyai banyak tujuan seperti diperdagangkan, dipakai, dipergunakan atau dimanfaatkan oleh konsumen.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa persediaan barang adalah barang-barang yang dimiliki untuk dijual kembali atau digunakan untuk menghasilkan barang yang akan dijual atau digunakan untuk keperluan pribadi.

2.2.3 Metode *First In First Out* (FIFO)

Metode FIFO menurut Lestari, Subagyo, dan Limantara (2019:28) merupakan metode dimana barang pertama yang masuk berarti barang tersebutlah yang pertama keluar. Dengan metode FIFO, biaya persediaan dihitung berdasarkan asumsi bahwa barang akan dijual atau dipakai sendiri dan sisa dalam persediaan menunjukkan pembelian atau produksi yang terakhir”.

Menurut Sari (2018:4), FIFO salah satu teknik pengelolaan queue atau penanganan tugas yang menumpuk, yaitu item yang pertama akan dikerjakan terlebih dahulu.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa Metode *First In First Out* (FIFO) adalah teknik dalam pengelolaan barang dimana barang yang pertama masuk akan dikelola terlebih dahulu.

2.2.4 Aplikasi Persediaan Barang Menggunakan Metode *First In First Out* (FIFO) Pada CV. Aneka Tunas Mandiri

Aplikasi Persediaan Barang menggunakan Metode *First In First Out* (FIFO) pada CV. Aneka Tunas Mandiri merupakan suatu aplikasi yang digunakan untuk mengelola data persediaan barang menggunakan metode *First in First Out* (FIFO) dimana barang yang pertama masuk ke gudang akan diberikan kepada pembeli terlebih dahulu.



2.3 Teori Khusus

2.3.1 *Unified Modeling Language* (UML)

Hutahaean dan Azhar (2018:21) berpendapat bahwa *Unified Modeling Language* (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak.

Nurhadiyan dan Junaedi (2019:39) mengatakan bahwa UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat tool untuk mendukung pengembangan sistem tersebut.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *Unified Modeling Language* (UML) adalah teknik untuk pemodelan desain program dan aplikasinya.

2.3.1.1 *Use Case Diagram*

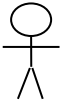
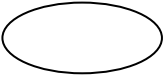
Use Case Diagram menurut Hutahaean dan Azhar (2018:21), merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat”.

Menurut Firdaur dan Saputra (2018:181), *Use Case Diagram* menjelaskan manfaat dari aplikasi jika dilihat dari sudut pandang orang yang berada diluar sistem (aktor).


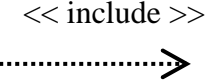
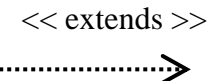
Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *Use Case Diagram* adalah pemodelan mengenai gambaran kelakuan (*behavior*) dari sistem informasi yang akan dibuat.

Adapun simbol-simbol dari *Use Case Diagram*.

Tabel 2.1 Simbol-simbol *Use Case Diagram*

No.	Simbol	Nama	Keterangan
1.		Actor	Seseorang atau apa saja yang berhubungan dengan sistem yang sedang dibangun
2.		Use case	Menggambarkan bagaimana seseorang menggunakan sistem

Lanjutan Tabel 2.1 Simbol-simbol *Use Case Diagram*

No.	Simbol	Nama	Keterangan
3.		Relasi asosiasi	Relasi yang dipakai untuk menunjukkan hubungan antara aktor dan use case
4.		Relasi include	Memungkinkan satu use case menggunakan fungsionalitas yang disediakan oleh use case lainnya
5.		Relasi extend	Memungkinkan suatu use case secara optional menggunakan fungsionalitas yang disediakan oleh use case lainnya

(Sumber : Rusmawan, 2019:72-73)

2.3.1.2 Activity Diagram


Hutahaean dan Azhar (2018:21) berpendapat bahwa *Activity Diagram* menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis.

Menurut Firdaus & Saputra (2018:182) mengatakan bahwa *Activity diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan oleh aktor, jadi aktivitas yang dapat dilakukan oleh system.



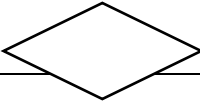

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *Activity Diagram* adalah gambaran aktivitas atau aliran kerja dari sebuah sistem.

Adapun simbol-simbol yang sering digunakan dalam *Activity Diagram* dapat dilihat pada tabel berikut ini:

Tabel 2.2 Simbol-simbol *Activity Diagram*

No.	Simbol	Nama	Keterangan
1.		Start state	Titik Awal atau permulaan

Lanjutan Tabel 2.2 Simbol-simbol *Activity Diagram*

No.	Simbol	Nama	Keterangan
2.		End state	Titik akhir atau akhir dari aktivitas
3.		Activity	<i>Activity</i> atau aktivitas yang dilakukan oleh actor
4.		Decision	Pilihan untuk mengambil keputusan
5.		Interaction	Alur

(Sumber : Rusmawan, 2019:80)

2.3.1.3 *Sequence Diagram*


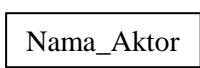
Hutahaean dan Azhar (2018:21) mengatakan bahwa *Sequence Diagram* menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

Menurut Firdaus & Saputra (2018:182) mengemukakan bahwa *Sequence diagram* adalah suatu diagram yang memperlihatkan atau menampilkan interaksi-interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu.


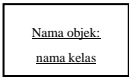
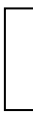
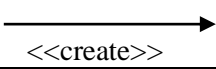

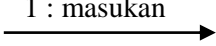

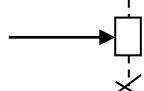
Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *Sequence Diagram* adalah diagram yang menunjukkan hubungan antar objek yang dijelaskan dengan rangkaian waktu.

Berikut adalah simbol-simbol yang digunakan pada *Sequence Diagram*:

Tabel 2.3 Simbol-simbol *Sequence Diagram*

No.	Simbol	Deskripsi
1.	<p>Aktor</p>  <p>Atau</p>  <p>Tanpa waktu aktif</p>	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tetapi aktor belum tentu berupa orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor

Lanjutan Tabel 2.3 Simbol-simbol *Sequence Diagram*

No.	Simbol	Nama
2.	Garis hidup/ <i>lifeline</i> 	Menyatakan kehidupan suatu objek
3.	Objek 	Menyatakan objek yang berinteraksi pesan
4.	Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya
5.	Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
6.	Pesan tipe <i>call</i> 1 : nama_metode() 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri
7.	Pesan tipe <i>send</i> 1 : masukan 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
8.	Pesan tipe <i>return</i> 1 : keluaran 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian
9.	Pesan tipe <i>destroy</i> <<destroy>> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy

(Sumber : Mukhtar, 2019:84-85)

2.3.1.4 Class Diagram

Diagram Kelas (*Class Diagram*) menurut Hutahaean dan Azhar (2018:21), merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

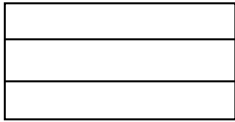


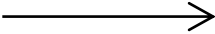
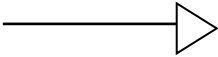
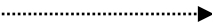
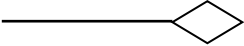
Menurut Firdaus & Saputra (2018:182), *Class diagram* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek.



Berdasarkan pengertian tersebut, dapat disimpulkan bahwa *Class Diagram* (Diagram Kelas) adalah gambaran mengenai kelas-kelas dalam sebuah sistem beserta hubungannya antara satu kelas dengan kelas lainnya.

Berikut adalah simbol-simbol yang digunakan pada *Class Diagram*:

Tabel 2.4 Simbol-simbol *Class Diagram*

No.	Simbol	Deskripsi
1.	<p>Kelas</p> 	Kelas pada struktur sistem
2.	<p>Antarmuka/<i>Interface</i></p>  <p>nama_interface</p>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
3.	<p>Asosiasi/<i>association</i></p> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
4.	<p>Asosiasi berarah/ directed association</p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
5.	<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
6.	<p>Kebergantungan/ dependency</p> 	Relasi antar kelas dengan makna kebergantungan antarmuka
7.	<p>Agregasi/<i>aggregation</i></p> 	Relasi antar kelas dengan makna semua bagian (whole-part)

(Sumber : Mukhtar, 2019:85)



2.4 Teori Program

2.4.1 PHP

Sulistiono (2018:5) berpendapat bahwa PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang digunakan untuk membuat website atau situs dinamis dan menangani rangkaian bahasa pemrograman antara *client side scripting* dan *server side scripting*.

Abdulloh (2018:127) mengatakan bahwa PHP merupakan kependekan dari *Hypertext Preprocessor* yaitu bahasa pemrograman web yang dapat disisipkan di skrip HTML dan bekerja di sisi server.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa PHP atau *Hypertext Preprocessor* adalah bahasa pemrograman yang digunakan untuk membuat website yang bekerja di antara sisi server dan sisi klien.

2.4.2 MySQL

Rusmawan (2019:97) menyebutkan bahwa MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*).

Sriwahyuni, Oktaria, dan Parma (2019:96) mengatakan bahwa MySQL merupakan *database* yang dikembangkan dari Bahasa SQL (*Structure Query Language*). SQL sendiri merupakan bahasa yang terstruktur yang digunakan untuk interaksi antara *script* program dengan *database server* dalam hal mengolah data.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa MySQL adalah sebuah tempat penyimpanan data yang mudah digunakan dan dibutuhkan saat ini untuk sebuah website secara gratis.

2.4.3 CodeIgniter

Sulistiono (2018:7) mengungkapkan bahwa Codeigniter adalah sebuah aplikasi *open source* yang berupa kerangka kerja atau *framework* untuk membangun website menggunakan bahasa pemrograman PHP. Tujuannya memungkinkan pengembangan proyek yang lebih cepat daripada penulisan kode



dasar atau kode terstruktur, dengan menyediakan banyak *library* yang biasanya digunakan dalam pengerjaan.

Hermanto dan Suyudi (2018:22) mengemukakan bahwa CodeIgniter merupakan web application network yang digunakan untuk membangun aplikasi php dinamis. CodeIgniter menjadi sebuah framework PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa Codeigniter adalah sebuah aplikasi berupa *framework* untuk membantu dalam membangun sebuah aplikasi web dengan lebih cepat.

2.4.4 XAMPP

XAMPP menurut Haqi dan Setiawan (2019:8), adalah perangkat lunak bebas (*free software*) yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program.

Menurut Mawaddah dan Fauzi (2018:2), XAMPP adalah *software web server* apache yang di dalamnya tertanam server MySQL yang didukung dengan bahasa pemrograman PHP untuk membuat website yang dinamis. XAMPP sendiri dapat berjalan pada berbagai macam platform seperti Windows, Linux, Mac OS X dan Solaris.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa XAMPP adalah sebuah *tool* yang menyediakan beberapa perangkat lunak seperti MySQL, PHP, dan perangkat lunak lainnya yang mendukung dalam membuat website.

2.4.5 HTML

Sulistiono (2018:2) mengatakan bahwa HTML (*HyperText Markup Language*) adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi seperti gambar, teks, video dan suara pada penjelajah web Internet, yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi.



Abdulloh (2018:7) mengungkapkan bahwa HTML merupakan singkatan dari *HyperText Markup Language* yaitu bahasa standar web yang dikelola penggunaannya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun setiap elemen dari website.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa HTML atau *HyperText Markup Language* adalah bahasa web yang digunakan untuk membuat halaman web berupa tag-tag yang menyusun setiap elemen dari halaman tersebut seperti gambar, text, video, maupun suara.

2.4.6 CSS

Sulistiono (2018:2) menjelaskan bahwa CSS (*Cascading Style Sheet*) merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. CSS bukan bahasa pemrograman. Sama halnya seperti style dalam aplikasi pengolahan kata, Microsoft Word yang dapat mengatur beberapa *style*, misalnya *heading*, *subbab*, *bodytext*, *footer*, *images*, dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (*file*).

Abdulloh (2018:45) berpendapat bahwa CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen web yang berfungsi mengatur elemen HTML dengan berbagai properti yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa CSS atau *Cascading Style Sheet* merupakan aturan untuk mengendalikan elemen-elemen HTML dengan beberapa *style* seperti *header*, *footer*, *bodytext*, dan *style* lainnya agar dapat digunakan disemua halaman web.

2.4.7 Bootstrap

Sulistiono (2018:17) mengatakan bahwa Bootstrap adalah sebuah pustaka *open source* yang merupakan *framework* CSS dan *Javascript* untuk membuat website yang responsif.



Hermanto dan Suyudi (2018:24) mengemukakan bahwa Bootstrap adalah sebuah *library framework* CSS yang dibuat khusus untuk bagian pengembangan *front-end* website.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa bootstrap merupakan sebuah *framework* CSS dan *Javascript* yang digunakan untuk mendesain sebuah web.

2.4.8 JavaScript

Abdulloh (2018:193) mengatakan bahwa JavaScript merupakan bahasa pemrograman web yang pemrosesannya dilakukan di sisi *client*. Karena berjalan di sisi *client*, JavaScript dapat dijalankan hanya dengan menggunakan browser.

Sulistiono (2018:4) mengungkapkan bahwa Javascript adalah bahasa pemrograman tingkat tinggi untuk membuat website dinamis, dan JQuery adalah *library* atau pustaka dari JavaScript yang dirancang untuk memudahkan penerapan *client side scripting* dan menyajikan sebuah paradigma baru pada penanganan *event* pada JavaScript.

Berdasarkan pernyataan tersebut, dapat disimpulkan bahwa javascript merupakan bahasa pemrograman untuk membuat website yang hanya dapat dijalankan dengan menggunakan browser.

2.5 Referensi Penelitian Sebelumnya

Berdasarkan penelitian yang dilakukan oleh Zaki, Samaji, dan Kusnadi pada tahun 2018, dalam jurnal yang berjudul “Perancangan Sistem Informasi Akuntansi Untuk Pengelola Persediaan Barang Dagang pada Pangkalan Kayu Tunas Harapan Ngamprah Bandung Barat”, ISSN : 2442-5826, salah satu permasalahan yang terjadi di Pangkalan Kayu Tunas Harapan yaitu pada pencatatan persediaan barang yang belum efektif sehingga Pangkalan Kayu Tunas Harapan kesulitan dalam mengetahui jumlah persediaan barang. Peneliti memutuskan untuk menggunakan metode *First In First Out* (FIFO) dalam pencatatan persediaan barang.



Berdasarkan penelitian yang dilakukan oleh Rusmawan pada tahun 2018, dalam jurnal yang berjudul "Sistem Informasi Persediaan Barang Menggunakan Metode *First In First Out*", ISSN : 2355-3421, penggunaan metode *First In First Out (FIFO)* dalam pembangunan sistem informasi akuntansi persediaan barang dapat membantu proses pengelolaan persediaan barang yang berjalan terutama dalam pembuatan data penjualan, data persediaan barang dan laporan bulanan menjadi terkomputerisasi sehingga dapat lebih cepat dan efektif dalam pengelolaan data persediaan barang.

Berdasarkan penelitian yang dilakukan oleh Fauziah dan Ratnawati pada tahun 2018, dalam jurnal yang berjudul "Penerapan Metode FIFO Pada Sistem Informasi Persediaan Barang", ISSN : 2442-2436, PT Fivalco Indonesia sering mengalami kelebihan persediaan barang, sehingga mengakibatkan menumpuknya stok barang digudang dan mengakibatkan kerusakan barang yang disimpan dalam gudang terlalu lama sehingga peneliti memutuskan untuk merancang sebuah sistem informasi persediaan barang dengan metode FIFO menggunakan aplikasi *Microsoft Visual Basic*.

Penelitian yang dilakukan oleh Monalisa, Putra, dan Kurnia pada tahun 2018, dalam jurnal yang berjudul "Rancang Bangun Sistem Informasi Inventory Obat Pada Rumah Sakit Jiwa Tampan Berbasis Web", ISSN : 2579-5341, peneliti menggunakan metode *First In First Out* untuk menghindari *expired* dari obat-obatan tersebut. Dengan FIFO, biaya inventori diperhitungkan dalam barang yang siap dijual atau dikonsumsi yang sudah ada lebih lama dan hal itu berarti stock yang tersedia adalah pembelian yang paling lama atau paling dulu diproduksi dan unit yang digunakan akan dibebankan pada harga dari barang yang terlama.

Berdasarkan penelitian yang dilakukan oleh Isbani, Juniadi, dan Setyobudi pada tahun 2018 dalam jurnal yang berjudul "Penerapan Aplikasi Akuntansi Persediaan Barang Dagang Dengan Metode FIFO pada Toko Annisa Boyolali", ISSN : 2503-3565, Toko Annisa Boyolali sering mengalami kesalahan pencatatan persediaan antara barang masuk dan barang keluar, tidak jarang pula memiliki persediaan yang berlebih sehingga peneliti memutuskan untuk membantu Toko



Annisa Boyolali dengan membuat komputerisasi persediaan barang menggunakan metode *First In First Out*.